REMARKS/ARGUMENTS

Applicants acknowledge the rejection of claims 1, 2, 4-10, 12, and 14-23 with a right to traverse. Claims 1, 2, 4, 5, 7-10, 12, 15, 16, 18, 19, and 20 are currently amended. The amendments to claims 1, 4, 5, 7-10, 12, 15, 16, and 18-20 are made merely to clarify the language of these claims, do not narrow or otherwise limit the scope of the claims, and are not made for any purpose related to patentability. No new matter has been introduced by way of any of the amendments. As a result, Claims 1, 2, 4-10, 12, and 14-23 are now pending in this application. Applicants respectfully request further examination and reconsideration of the rejections for the reasons stated below.

Claim Rejections - 35 U.S.C. §103

Independent claims 1, 10 and 12 were rejected as being unpatentable over U.S. Patent No. 5,371,878 (hereinafter "Coker") in view of "How Debuggers Work" by Jonathan B. Rosenberg (hereinafter "Rosenberg). Applicants respectfully traverse in view of the following.

Claim 1 recites a boot method for synchronizing a microcontroller and a virtual microcontroller of an In-Circuit Emulation system in lock step through executing a set of boot code in the microcontroller and executing a dummy code in the virtual microcontroller, wherein the dummy code is a set of timing code to take the same number

of clock cycles as the microcontroller and wherein the set of boot code is inaccessible to the virtual microcontroller.

Paragraph 5 of the Office Action mailed on September 17, 2007 states that Applicants' specification does not provide a deliberate and explicit definition for the term "dummy code." On this comment, Applicants respectfully bring to attention relevant clauses in MPEP 2111.01 that "[d]uring the examination, the claims must be interpreted as broadly as their terms reasonably allow." *In re American Academy of Science Tech Center, 367 F.3d 1359, 1369 70 USPQ2d 1827, 1834 (Fed. Cir. 2004).* This means that the words of the claim must be given their plain meaning where the plain meaning refers to the ordinary and customary meaning given to each one of the words. *In re Zletz, 893 F2d 319, 321, 13 USPQ2d 1320, 1322 (Fed. Cir. 1989).* The term "dummy code" is used in the instant specification and claims.

In constructing a claim term, dictionary meanings must be compared against the use of the term in context, and the intrinsic record must be always consulted to identify which of the different possible dictionary meanings is most consistent with the use of the word by the inventor. *ACTV, Inc. v. The Walt Disney Company, 346 F.3d 1082, 1092, 68 USPQ2d 1516, 1524 (Fed. Cir. 2003).*

Unabridged Dictionary defines "dummy" within the confines of computers as "an artificial address, instruction, or other datum fed into a computer only to fulfill prescribed conditions and not affecting operations for solving problems." Furthermore, American Heritage Dictionary defines "dummy" within the confines of computer science as "a

character and other piece of information entered into a computer only to meet prescribed condition, such as word length, and having no affect on operations."

Dictionary meanings of "dummy code" clearly show that the ordinary and customary meaning of "dummy code" is different from "booting code" recited in Claim 1. Unlike "booting code" which is used to boot the microcontroller, "dummy code" does not perform any functions to boot the virtual microcontroller. "Dummy code" is used in one embodiment to synchronize the virtual controller with the microcontroller using a timing loop.

The specification usage of this term is consistent with the dictionary meanings of the term. This is evidenced by the following passage:

> "[the] boot code used by microcontroller 232 to carry out various initialization process may also contain proprietary information which should not be exposed in the clear. For example, serial numbers, passwords, and algorithms embedded within the boot code for carrying out various initialization operations may be contained in the boot code. Thus, it is advantageous that this code is kept protected within the confines of the microcontroller 232..." "The above process for booting the real microcontroller 232 and virtual microcontroller 220 in synchronization permits not only synchronized operation, but it does so in a manner that protects any proprietary information that might be stored within the real microcontroller 232's boot code. Since only dummy code runs in the virtual microcontroller 220, there is no reason to expose any proprietary boot code stored in the real microcontroller 232 simply to accomplish synchronization." See Page 26, Line 15 to Page 28, Line 6 of Applicants' specification.

Since "dummy code" according to the passage in Applicants' specification does not contain proprietary information which includes the serial numbers, passwords and algorithms of the boot code, it is consistent with the dictionary meanings of the term. Thus, the dictionary meanings of the term should be interpreted as what the Applicants intended for the term, and "dummy code" recited in Claim 1 is not same or similar as boot code since "dummy code" is different from boot code in substance.

Paragraph 5 of the Office Action further states that the "inaccessibility" of the microcontroller code has been addressed in the Previous Office Action, at pages 4-5. The Previous Office Action mailed on April 2, 2007 states the following:

> "The text of the Coker reference plainly discloses that there are differences between the operation of the target-ECS and the shadow system, and therefore the software code executed by the target-ECS and the shadow system, as explained above. Further, presuming *arguendo* that the target-ECS and the shadow system execute precisely the same code, this in no way suggests that the target-ECS code is accessible to the shadow system. For the target-ECS code to be accessible to the shadow system, the Coker reference would need to disclose or otherwise suggest that a mechanism actually exists in the Coker invention whereby the shadow system can read from, write to, or receive by transmission the target-ECS code. There is no such disclosure in the Coker reference."

Applicants respectfully traverse, and understand Coker to provide such disclosure. Coker explicitly states that "[t]he shadow system receives its input data from the input registers of the target-ECS and stores the input data in its RAM," See Column 2, Lines 61-63 (emphasis added) and that "[t]he target-ECS is connected by a one-directional bus with an interface means comprising means for receiving input signals from the target-

ECS and converting the input signals to generic input signals, means for converting the generic input signals to unique input events and providing said unique input events at the first opportunity to a connected shadow system or host system." See Column 4, Lines 9-16. The passages in Coker at least indicate that the target-ECS may be accessible by the shadow system. Unlike the limitation of Claim 1, Coker does not teach that the target-ECS is inaccessible by the shadow system. Instead, Coker teaches that the target-ECS can be accessed by the shadow system throughout their operations.

Accordingly, Applicants respectfully disagree with Paragraph 4 of the rejection that "Applicants' disclosure and claimed invention describes two systems executing the same or similar software from startup or reset." Since at least the two limitations of Claim 1 that "the dummy code" is different from "the boot code" and that "the boot code" is inaccessible by the virtual microcontroller are different from the prior art, Applicants respectfully request the withdrawal of the rejection.

Independent claims 10 and 12 recite at least the two limitations similar to that of Claim 1 and are therefore patentable over the cited references for the same reasons. As such, allowance of independent Claims 10 and 12 is earnestly solicited.

With respect to remaining Claims that depend on Claims 1, 10 and 12, Applicants respectfully assert that the Claims overcomes the rejections of record for at least the rationale previously presented with respect to the independent Claims, and respectfully solicit allowance of these Claims.

In addition with respect to currently amended Claim 2, Applicants respectfully assert that Rosenberg or Coker or their combination fails to teach or fairly suggest the limitation of "after completion of the simultaneous halting" as recited by Claim 2. Applicants respectfully assert that the combined references are completely silent as to the limitation. Applicants also respectfully submit that no new matter has been added in the claim and that the claim is fully supported in the application. Support for Claim 2 is found throughout the document and at least where it is described; "...The In-Circuit Emulation system then issues a Halt command at 628 to cause the microcontroller to halt. At this point the In-Circuit Emulation system copies the register contents and memory contents from the real microcontroller 232 to the virtual microcontroller 220...," See Page 27, Lines 17-20. For this additional reason, Applicants respectfully assert that Claim 2 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to Claim 4, Applicants respectfully assert that Rosenberg or Coker or their combination fails to teach or suggest the limitation that the microcontroller and the virtual microcontroller are pointing to the same assembly instruction line 0 after the executing the boot code. Applicants respectfully assert that the combined references are completely silent as to the limitation. For this additional reason, Applicants respectfully assert that Claim 4 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to Claim 5, Applicants respectfully assert that Rosenberg or Coker or their combination fails to teach or suggest the limitation that a breaker is set at an assembly instruction line 0 prior to the executing the boot code, and prior to the executing

the timing code. Although Rosenberg teaches that "[a] break point is a special code placed in the executing code stream," Rosenberg does not teach where and when the breakpoint is set. Applicants respectfully assert that the combined references are completely silent as to the limitations. For this additional reason, Applicants respectfully assert that Claim 5 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to Claim 7, Applicants respectfully assert that the Claim combines the limitations of Claim 4 and 5. For the reasons stated for the two Claims, Applicants respectfully assert that Claim 7 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to Claim 8, Applicants respectfully assert that the Claim combines the limitations of Claim 2, 4 and 5. For the reasons stated for the three Claims, Applicants respectfully assert that Claim 8 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to currently amended Claim 20, Applicants respectfully assert that Rosenberg or Coker or their combination fails to teach or suggest the limitation of removing the breaker at assembly instruction line 0 after the copying the register contents and the copying the memory contents." The references do not teach where and when the breakpoint is set. Applicants respectfully assert that the combined references are completely silent as to the limitations. For this additional reason, Applicants respectfully

assert that Claim 20 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

With respect to previously presented Claim 22, Applicants respectfully assert that Rosenberg, Coker, Matyas or their combination fails to teach or suggest the limitation of "the boot code containing algorithms." The rejection states that Matyas teaches code that comprises serial number, passwords, and algorithms "[i]n carrying out the computer/smart card protocol, the T output is sent to the smart card together with the parameter P1 (password) and P2 (programming number | diskette serial number, where | denotes concatenation) and a third parameter P3" where P3 is the computer number and the DES algorithm is used to encrypt the file key. Although Matyas teaches the passing of serial number and password as a part of the code being passed, the reference does not teach that any algorithm is contained in the code being passed. The third parameter P3 being passed is a computer number interpreted by the DES algorithm, See Column 13, Lines 13-15. Matyas does not teach that the DES algorithm, or any other algorithm, is passed along with the serial number and password as a part of the code. For this additional reason, Applicants respectfully assert that Claim 22 overcomes the rejections of record, and respectfully solicit allowance of this Claim.

For the reasons stated above, Applicants earnestly solicit the allowance of Claims 1, 2, 4-10, 12 and 14-23.

CONCLUSION

In light of the above listed remarks, reconsideration of the rejected Claims is requested. Based on the arguments presented above, it is respectfully submitted that Claims 1-2, 4-10, 12 and 14-23 overcome the rejections of record and, therefore, allowance of Claims 1-2, 4-10, 12 and 14-23 is earnestly solicited.

Please charge any additional fees or apply any credits to our PTO deposit account number: 23-0085

Respectfully submitted,

Murabito, Hao & Barnes LLP

Date 10/31/2007 By 

Steve S. Ko
Reg. No. 58,757

Two North Market Street
Third Floor
San Jose, CA 95113
(408) 938-9060